On the intertwining between capacity scaling and TCP congestion control

# On the Intertwining between Capacity Scaling and TCP Congestion Control

Carla Panarello
CNIT - Unit Research of
University of Catania - Italy
cpana@dieei.unict.it

Marco Ajmone Marsan
DET
Politecnico di Torino - Italy
IMDEA Networks - Spain
ajmone@tlc.polito.it

Alfio Lombardo
DIEEI
University of Catania - Italy
lombardo@dieei.unict.it

Marco Mellia
DET
Politecnico di Torino - Italy
mellia@tlc.polito.it

Michela Meo
DET
Politecnico di Torino - Italy
meo@tlc.polito.it

Giovanni Schembra
DIEEI
University of Catania - Italy
schembra@dieei.unict.it

## ABSTRACT

Recent works advocate the possibility of improving energy efficiency of network devices by modulating switching and transmission capacity according to traffic load. However, addressing the trade-off between energy saving and Quality of Service (QoS) under these approaches is not a trivial task, specially because most of the traffic in the Internet of today is carried by TCP, and is hence adaptive to the available resources.

In this paper we present a preliminary investigation of the possible intertwining between capacity scaling approaches and TCP congestion control, and we show how this interaction can affect performance in terms of both energy saving and QoS.

## Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Network communications*; C.4 [**Performance of Systems**]: Performance attributes; Reliability, availability, and serviceability

## General Terms

Performance

## 1. INTRODUCTION

Providing network devices with power management capabilities which can optimize the tradeoff between power saving and quality of service (QoS) is a key goal for the successful development of green networking technologies. Recent works advocate the possibility of improving energy efficiency of network devices by modulating switching and transmission capacity according to their traffic load [1, 2, 3, 4]. However, addressing the trade-off between energy saving and QoS under these approaches is a non-trivial task.



To assess whether the reduction of the overall network energy consumption can be achieved without adversely affecting network performance, most of the schemes proposed in the literature were tested using real-world network topologies and traffic workloads. However, up to now, to the best of our knowledge, traffic workloads were reproduced by artificially generating traffic according to measured patterns or statistics, but ignoring the feedback that energy saving schemes may have on the traffic itself. That is, without taking into account the possibility that changes in the instantaneous offered traffic rate may be caused by the effect that energy-saving mechanisms have on rate controlled sources, such as TCP. In fact, on the one hand, energy-saving mechanisms modify the amount of resources available at network devices according to traffic, but, on the other hand, TCP traffic sources adapt their sending rate according to the available resources. Therefore, unexplored looped reactions occur when the TCP congestion control mechanism is coupled with power management capabilities.

Since most of the traffic in today's Internet is carried by TCP, the study of this intertwining is of fundamental importance for energy-efficient networking research.

In this paper we present a preliminary exploration of the interactions between the congestion control of TCP and capacity scaling approaches. Simulation results for simple scenarios show that, in most of the considered cases, non-negligible energy savings are achieved at the expense of a decreased QoS, measured as completion time for web downloads. Indeed, the performance of TCP traffic aggregates is badly affected by power control mechanisms implemented at routers through capacity scaling.

We emphasize that the results presented in this paper are partial and preliminary. A much more detailed analysis is necessary to understand the trade-off between energy saving and QoS for TCP traffic.

## 2. PROBLEM DEFINITION AND CASE STUDY

Optimizing the tradeoff between power saving and QoS is a challenging research objective, whose difficulty is rooted in the dominant presence, in the current Internet, of TCP traffic, and in its elastic nature. More specifically, the TCP congestion control mechanism makes the source sending rate variable over time, so as to adapt to the available network resources. This translates into a variable input traffic at routers, which activate their power management schemes, trying to scale the transmission capacity accordingly. In turn, the variation of resources at routers determined by power management schemes induces changes in the network available resources, which affect TCP congestion control.

The investigation of the behavior of the resulting closed loop is the objective of this work.

To explore these issues, we have conducted extensive simulations for different conditions. In particular, we have considered a simple, yet realistic scenario in which traffic load is generated according to the statistics of actual web traffic. We have collected TCP flow length statistics using Tstat[5] for one entire day at the Politecnico di Torino edge campus link. The flow length probability distribution is then used in simulations. The flow arrival process is modeled as a simple Poisson process, whose rate is computed so as to control the average offered traffic rate at the bottleneck link.

We consider a power management mechanism based on the scheme proposed in [4], and there referred as practRA. In summary, this technique uses the history of packet arrivals to predict the future arrival rate at time $t$[1], $\hat{r}_f$, which is estimated with an exponentially weighted moving average (EWMA) of the measured history of past arrivals. The current buffer size $q$ and the current node service rate $r_i$ are used to estimate the potential queuing delay, so as to avoid violating a given delay constraint, $d$. The goal of the algorithm is to determine the service rate closest to the arrival rate, among a set of available values. A rate transition can occur only if at least $\Delta$ seconds passed since the previous transition. Additionally, the device cannot send packets for $\delta$ seconds after each transition ($\delta < \Delta$).

The algorithm works as follows:

- A link operating at rate $r_i$ with current queue size $q$ increases its rate to $r_{i+1}$ iff $\left(\frac{q}{r_i} > d \text{ OR } \frac{\delta \hat{r}_f + q}{r_{i+1}} > d - \delta\right)$;
- A link operating at rate $r_i$ with current queue size $q$ decreases its rate to $r_{i-1}$ iff $\left(q = 0 \text{ AND } \hat{r}_f < r_i\right)$.

The evaluation of the impact of the capacity scaling mechanism on TCP performance is presented both in terms of potential energy saving and QoS. More specifically, as far as QoS is concerned, for web-like traffic we consider the completion time, i.e., the time needed for completing the download of the requested object. The potential energy saving is evaluated considering two possible technologies: Frequency Scaling (FS) and Dynamic Voltage Scaling (DVS) [6]. The first one allows the operating frequency to be scaled linearly with load. The second technology, implemented in devices operating at different frequencies, allows the operating voltage to be further regulated according to the operating frequency, thus scaling power consumption cubically with the operating frequency. FS and DVS have been successfully applied to general purpose processors (P-states in Intel processors).

For both FS and DVS, we assume that each network interface supports a number of states corresponding to different service rates. The power consumption of a device is a function of the service rate $\mu$. In addition, as typical for actual devices, a portion of the power consumption is assumed to be constant and independent of the operating rate. Therefore, the power $P(\mu)$ consumed by a network device can be computed as: $P(\mu) = P_s + f(\mu)$, where $P_s$ is the static amount of power and $f(\mu)$ represents the rate-dependent portion of consumption. For FS and DVS $f(\mu) = O(\mu)$ and $f(\mu) = O(\mu^3)$, respectively. For DVS, we consider an additional constraint because, in practice, there is a minimum rate below which scaling the link rate offers no further voltage reduction. Therefore, we introduce a maximum scaling factor $\phi$, such that $f(\mu) = O(\mu^3)$ for $\mu \in [\mu_{MAX}/\phi; \mu_{MAX}]$.

[1]To simplify notation, we omit the explicit indication of dependence on time $t$.

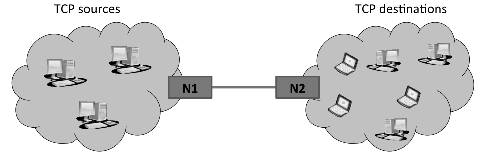



**Figure 1: Network topology**

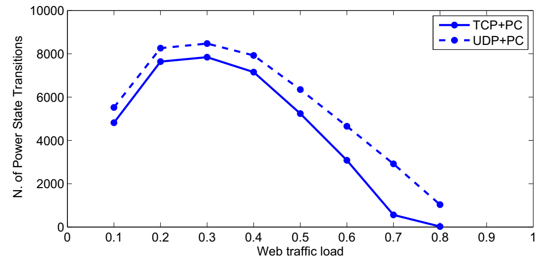



**Figure 2: Number of Power State Transitions**

We consider values of $P_s$ in $\{0, 0.25, 0.5, 0.75\}$ and $\phi = 2$.

# 3. NUMERICAL RESULTS

In this section we show the effects of the intertwining between the TCP congestion control and the capacity scaling approach to energy saving. To this purpose we conducted extensive simulations with the ns-2.30 simulator.

The topology used for simulations is depicted in Fig. 1. The duration of each simulation run is 600 seconds. The number of source nodes varies to model different load; the applications running on each node generate web-like traffic (only in the server-to-client direction), as previously described.

The MTU length is 1500 bytes. Sources use TCP NewReno. The average round-trip time is 50 ms. The buffer size of nodes $N1$ and $N2$ is set equal to the bandwidth-delay product, whereas the buffer size of remaining nodes is set large enough to be considered unlimited. The maximum capacity of the link between nodes $N1$ and $N2$ is set to 10 Mb/s. The capacity scaling mechanism is implemented in node $N1$, which controls the bottleneck link in the server-to-client direction. The set of available service rates for the capacity scaling algorithm is distributed in the range $[1 \div 10]$ Mb/s, with steps of 1 Mb/s. The transition time $\delta$ is equal to 2 ms, the inter-switch time, $\Delta$ is equal to 16 ms, and the delay constraint is equal to $d = 40$ ms.

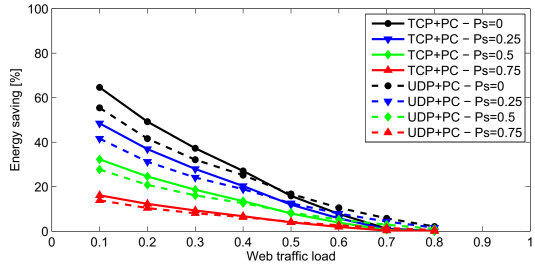
## 3.1 Impact of TCP on energy savings

In this subsection we show how the presence of the TCP congestion control mechanism affects the energy savings of the capacity scaling approach. To this purpose, we consider three scenarios. In the first one, named $TCP + PC$, node N1 implements the power control algorithm, and TCP is used at the transport layer; we thus have both TCP and power control in action.

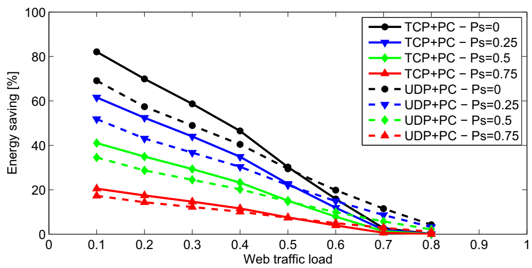
In the second scenario, named *TCPonly*, N1 does not implement power control, and TCP is the only control mechanism.

Finally, in the $UDP + PC$ scenario, N1 implements power control, and the aggregate of TCP sources is substituted by a single, uncontrolled UDP source, whose packet sending rate replays the same packet sending rate recorded under the *TCPonly* scenario; in this case, power control is in action alone, but the packet generation rate is the same as if TCP were used with no power control.

Fig. 2 shows the number of rate transitions requested by the




(a) FS




(b) DVS

**Figure 3: Comparison between the potential energy saving achieved by the capacity scaling technique for TCP and UDP traffic**

power control algorithm in the cases of TCP and UDP traffic. The difference between the two curves in the figure indicates that the nature of the traffic induces a different behavior of the capacity scaling algorithm.

Fig. 3 shows a comparison between the potential energy saving of the capacity scaling algorithm with respect to the *TCPonly* case, when the traffic loading the node is either TCP or UDP.

Simulation results demonstrate that the presence of TCP traffic modifies the capacity scaling performance. More specifically, for low values of traffic load, the presence of TCP allows more energy to be saved with respect to UDP. This is mainly due to the more controlled TCP sending rate. When some losses occur, the smoother arrival rate at node $N1$ reduces the loss probability (see also Fig. 4), increasing the possibility to save energy. Instead, when UDP traffic mimics the TCP sending rate, no reaction is present to losses, so the arrival rate at node $N1$ does not decrease during congestion periods, and the capacity scaling algorithm does not reduce the service rate, thus consuming more energy.

When the traffic load increases, the results in Fig. 3 show an opposite behavior. The average transmission window of TCP sources decreases, so the reduction of the sending rate due to TCP congestion control is more limited. Moreover, the high traffic load increases the loss probability, thus the number of retransmissions increases as well (Fig. 4). Overall, the arrival rate at node $N1$ increases, and the capacity scaling algorithm has little possibility to save energy. On the contrary, when UDP traffic is loading node $N1$, the high loss probability produces no further increase of the packet arrival rate, so the energy saving is misleadingly higher than in the $TCP+PC$ case.

## 3.2 Impact of Capacity scaling on TCP performance

The results of Section 3.1 demonstrate that the performance of the capacity scaling approach is affected by the nature of the traffic

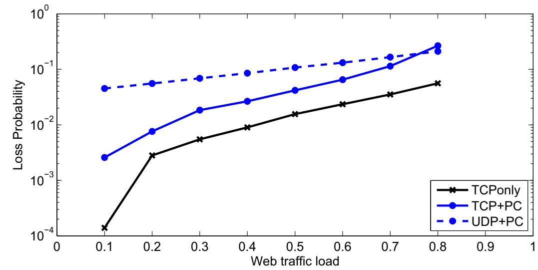



**Figure 4: Loss probability for TCP and UDP traffic**

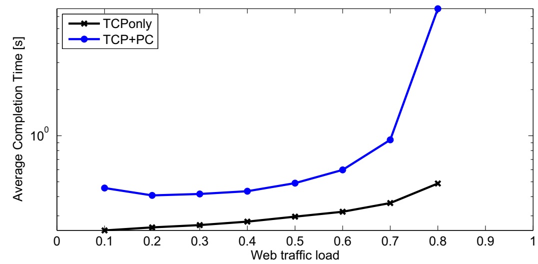



**Figure 5: Average completion time**

loading the node. In this section, we show that also the network performance is affected by the implementation of the capacity scaling approach.

Fig. 4 shows that the loss probability in the node implementing the capacity scaling algorithm is higher when power control is used ($TCP+PC$) with respect to the opposite case ($TCPonly$). This means that the capacity scaling algorithm reduces power consumption, but penalizes network performance. More specifically, as the traffic load increases, the interaction between the congestion control mechanism of TCP and power control highly increases losses.

To further evaluate the impact of the power control mechanism on QoS, Fig. 5 shows the average completion time of web downloads as a function of offered load. As expected, results show that the average transfer time increases with traffic load. However, when the capacity scaling technique is applied, the average transfer time is at least two times higher than obtained in the case $TCPonly$, reported as reference. These results tell us that capacity scaling may badly affect QoS.

For example, at offered load equal to 0.8, the completion time grows more than tenfold, from 0.2s to more than 7s!

## 3.3 Power control with TCP traffic: is it worth?

Results presented so far show that the interaction between capacity scaling and TCP congestion control affects the performance in terms of both QoS and energy saving. More specifically, Fig. 3 shows that large energy savings can be achieved with the capacity scaling approach for low traffic load, whereas QoS, in terms of average completion time for web downloads, suffers a tolerable degradation (Fig. 5). On the contrary, for high traffic load, no significant reduction in energy consumption is achievable, as expected. However, Fig. 5 shows that we are trading negligible energy consumption for unacceptable QoS degradation.

To summarize the system performance in terms of both QoS and energy saving with a single parameter, we show the difference between the percentage of energy saving and the percentage of QoS degradation with respect to the $TCPonly$ case. This parameter, in the following indicated as $ES-QoS_D$, scales the energy saving

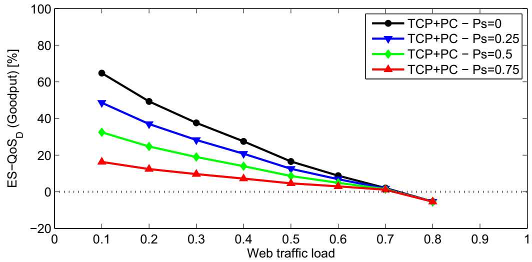



**Figure 6: Difference $ES-QoS_D$, when the QoS metric is the goodput**

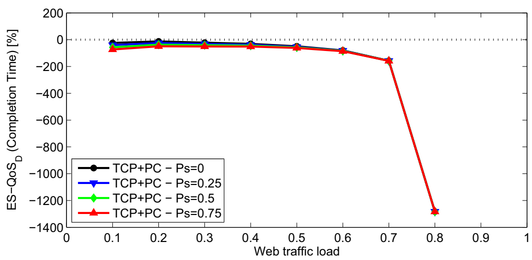



**Figure 7: Difference $ES-QoS_D$, when the QoS metric is the average completion time**

shown in Fig. 3 according to the reduction of QoS.

Note that, negative values of the parameter $ES-QoS_D$ are obtained when the percentage QoS degradation is higher than the percentage energy saving.

Fig. 6 and Fig. 7 show the $ES-QoS_D$ parameter for the FS technology, when either goodput or average web download completion time are considered as QoS metric, respectively. Similar results are obtained by considering the DVS technology.

When the goodput is taken as the QoS metric, the impact of capacity scaling on QoS is small. Yet, for high traffic load, the percentage QoS degradation is higher than the percentage energy saving, so that the difference $ES-QoS_D$ becomes lower than zero (see Fig. 6).

Instead, when the average completion time is considered as QoS metric, Fig. 7 shows that, for low traffic load, the difference $ES-QoS_D$ is close to zero, meaning that energy saving is possible, but almost the same percentage of QoS degradation is present. Instead, the difference $ES-QoS_D$ decreases and becomes largely lower than 0 for high traffic load, since the energy saving becomes very low and the QoS degradation very high, meaning that in this region no capacity scaling should be used.

# 4. CONCLUSIONS

This paper presented an initial exploration of the intertwining between power control mechanisms and TCP congestion control. Results demonstrate that mutual reactions exist between TCP and capacity scaling, and they affect performance in terms of both energy saving and QoS. More specifically, simulation results show that important energy savings are possible without significantly degrading QoS only when the traffic load is low. On the contrary, as traffic load increases, no significant energy saving is achieved, while QoS degradation becomes intolerable. As an example, when the offered load is about 0.8, the completion time for web downloads grows by over one order of magnitude.

Note that the results presented in this paper may be affected by the parameters of the capacity scaling algorithm, so that a more accurate study is needed to better understand the relationship between performance and design parameters. For example:

- the delay constraint $d$ used by the capacity scaling algorithm determines the average queue length allowed by the algorithm. Obviously, the higher the queue length is, the higher the loss probability is. Smaller values of $d$ make the algorithm more reactive, and, at the same time, force a high number of power state transitions, which may lead to the instability of power control;
- the reactivity of the algorithm is also measured by the choice of the $\Delta$ parameter, which represents the minimum time interval between two consecutive power state transitions. Higher values of $\Delta$ allow the number of power state transitions to be reduced, whereas smaller values allow the service rate to quickly adapt to variations of the traffic load;
- the granularity of the available service rates is expected to affect performance;
- the impact of the buffer size needs to be investigated, as well as the applicability in this context of the rules proposed in the literature for the choice of the buffer size when TCP performance is taken into account;
- a possible solution to allow the algorithm to react quickly to variations of the traffic load without increasing the number of power state transition, is to consider service rate transitions between non-consecutive values.

These considerations suggest that further investigations are necessary. Nevertheless, the analysis of the preliminary results presented in this paper clearly indicates that a careful design of power control mechanisms is mandatory in order to be able to reduce energy consumption without unacceptably compromising QoS.


# Acknowledgments

The research leading to these results was funded through the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n. 257740 - Network of Excellence TREND (Toward Real Energy-efficient Network Design).


# 5. REFERENCES


[1] R.Bolla, R.Bruschi, F.Davoli, F.Cucchietti, *Energy Efficiency in the Future Internet: A Survey of Existing Approaches and Trends in Energy-Aware Fixed Netweork Infrastuctures*, IEEE Communications Surveys & Tutorials, vol.13, no.2, pp.223-244, 2011.

[2] R.Bolla, R.Bruschi, F.Davoli, A.Ranieri, *Performance Constrained Power Consumption Optimization in Distributed Network Equipment*, Green Communications Workshop, Dresden, Germany, June 2009.

[3] A.Wierman, L.L.H.Andrew, A.Tang, *Power-Aware Speed Scaling in Processor Sharing Systems*, IEEE INFOCOM 2009, Rio de Janeiro, Brazil, April 2009.

[4] S.Nedevschi, L.Popa, G.Iannaccone, S.Ratnasamy, D.Wetherall, *Reducing Network Energy Consumption via Sleeping and Rate-Adaptation*, USENIX/ACM NSDI 08, San Francisco, USA, April 2008.

[5] A.Finamore, M.Mellia, M.Meo, M.Munafò, D.Rossi, *Experiences of Internet Traffic Monitoring with Tstat*, IEEE Network, vol.25, no.3, March/April 2011.

[6] B.Zhai, D.Blaauw, D. Sylvester , K. Flautner, *Theoretical and Practical Limits of Dynamic Voltage Scaling*, DAC 2004, San Diego, California, USA, June 2004.